

AMENDMENTS TO THE CLAIMS

This listing of the claims shall replace all prior versions and listing of the claims in this application:

1. (Amended) A method for initializing a storage controller, comprising:
 - commencing an initial microcode load (IML) operation;
 - initializing a parameter in a field of the tracks of copy state data to a first state;
 - commencing background staging of copy state data tracks from a disk storage device to a memory device;
 - changing the state of the parameter of a track to a second state;
 - receiving a request to access a track of copy state data;
 - if the requested track of copy state data has been staged and the parameter is in the second state, accessing the requested track of copy state data;
 - if the requested track of copy state data has not been staged and the parameter is in the first state:
 - issuing a wait command in response to the request to access the track of copy state data;
 - staging the requested track of copy state data;
 - revoking the wait command; and
 - accessing the requested track of copy state data;
 - completing the staging of the copy state data tracks; and
 - completing the IML.
2. (Cancelled)
3. (Cancelled)
4. (Amended) The method of claim 3~~1~~, wherein completion of the IML is independent of completion of the staging of all copy state data tracks.

5. (Original) A method for processing metadata in a data storage controller, comprising:

- executing a copy service operation;

- maintaining a current state of the copy services operation in a memory device;

- periodically destaging the current state of the copy services operation from the memory device to a plurality of metadata tracks on a storage device;

- following commencement of an error handling routine, commencing background staging of the metadata tracks from the storage device to the memory device;

- receiving a request to access a metadata track;

- if the requested metadata track has been staged, allowing access to the requested metadata track;

- if the requested metadata track has not been staged:

- issuing a wait command in response to the request to access the metadata track;

- staging the requested metadata track;

- revoking the wait command; and

- accessing the requested metadata track;

- completing the staging of the metadata tracks; and

- completing the error handling routine.

6. (Original) The method of claim 5, further comprising

- initializing a parameter in a field in each of the metadata tracks to a first state when the error handling routine is commenced; and

- changing the state of the parameter of a metadata track to a second state when the metadata is staged to the memory device.

7. (Original) The method of claim 6, further comprising, when the request to access the metadata track is received:

if the parameter is in the second state, allowing access to the requested track; and

if the parameter is in the first state:

staging the requested track; and

allowing access to the requested track.

8. (Original) The method of claim 5, wherein the error handling routine is an initial microcode load.

9. (Original) The method of claim 5, wherein completion of the error handling routine is independent of completion of the staging of metadata.

10. (Original) A method for processing metadata in a data storage controller, comprising:

executing a copy service operation;

maintaining a current state of the copy services operation in a memory device;

periodically destaging the current state of the copy services operation from the memory device to a plurality of metadata tracks on a storage device;

following commencement of an error handling routine, initializing a parameter in a field of each of the metadata tracks to a first state;

commencing background staging of the metadata tracks from the storage device to the memory device;

changing the state of the parameter of a metadata track to a second state when the metadata track is staged to the memory device.

receiving a request to access a metadata track;

if the parameter is in the second state, allowing access to the requested metadata track;

if the parameter is in the first state:

issuing a wait command in response to the request to access the metadata track;

- staging the requested metadata track;
- revoking the wait command; and
- allowing access to the requested metadata track; and

completing the error handling routine.

11. (Original) The method of claim 10, wherein the error handling routine is an initial microcode load.

12. (Original) The method of claim 10, wherein completing the error handling routine is independent of completion of the staging of metadata tracks.

13. (Original) A storage controller, comprising:

- means for receiving customer data from a host device;
- means for storing the customer data onto a first storage device;
- means for copying the customer data onto a second storage device in a copying operation;
- a memory device for storing a current state of the copying operation as a plurality of metadata tracks;
- means for periodically destaging the metadata tracks to a selected one of the first and second storage devices; and
- means for processing an error handling routine following an interruption in the copying operation, comprising:
 - means for initializing a parameter of each metadata track in the memory device to a first state;
 - means for commencing background staging of the metadata tracks from the selected storage device to the memory device;
 - means for changing the state of the parameter of a metadata track to a second state when the metadata track is staged to the memory device;
 - means for receiving a request to access a metadata track;

means for allowing access to the requested metadata track if the parameter of the requested metadata track is in the second state;

if the parameter of the requested metadata track is in the first state:

means for issuing a wait command in response to the request to access the metadata track;

means for staging the requested metadata track to the memory device;

means for revoking the wait command; and

means for allowing access to the requested metadata track;

and

means for completing the error handling routine.

14. (Original) The storage controller of claim 13, wherein the error handling routine is an initial microcode load.

15. (Original) The storage controller of claim 13, wherein the means for completing the error handling routine comprises means for completing the error handling routine independent of completion of the staging of metadata tracks.

16. (Original) A copy services component of a data storage controller, the copy services component comprising:

means for directing that customer data be copied onto a storage device in a copy operation;

a plurality of data structures for collectively maintaining a current state of the copy operation;

an interface through which copies of the data structures are periodically destaged to the storage device;

means for processing an error handling routine following an interruption in the copy operation, comprising:

means for initializing a parameter of each metadata track in the memory device to a first state;

means for commencing background staging of the metadata tracks from the selected storage device to the memory device;

means for changing the state of the parameter of a metadata track to a second state when the metadata track is staged to the memory device.

means for receiving a request to access a metadata track;

means for allowing access to the requested metadata track if the parameter of the requested metadata track is in the second state;

if the parameter of the requested metadata track is in the first state:

means for issuing a wait command in response to the request to access the metadata track;

means for staging the requested metadata track to the memory device;

means for revoking the wait command; and

means for allowing access to the requested metadata track;

and

means for completing the error handling routine.

17. (Original) The copy services component of claim 16, wherein the error handling routine is an initial microcode load.

18. (Original) The copy services component of claim 16, wherein the means for completing the error handling routine comprises means for completing the error handling routine independent of completion of the staging of metadata tracks.

19. (Original) A data structure stored in a memory of a data storage controller, the storage controller coupled to a first storage device storing customer data and to a second storage device storing a copy of the customer data, the data structure comprising:

a first field for storing a portion of a current state of an active copy operation, the portion being periodically destaged to one of the first and second storage devices; and

a track state field having a first state indicative of invalid contents in the first field and a second state indicative of valid contents in the first field;

wherein:

following commencement of an error handling routine, a background staging commences of the track from the storage device to the memory device;

when a request is received to access the first field:

if the first field has been staged, access is allowed to the track;

if the first field has not been staged:

a wait command is issued in response to the request to access the first field;

the first field is staged;

the wait command is revoked; and

the first field is accessed; and

the error handling routine is completed.

20. (Original) The data structure of claim 19, wherein further:

the track state field is initialized to the first state when the error handling routine is commenced; and

the state of the track state field is changed to the second state when the first field is staged to the memory device.

21. (Original) The data structure of claim 20, wherein further, when the request to access a track is received:

if the track state field is in the second state, access to the first field is allowed; and

if the track state field is in the first state:

the first field is staged; and
access to the first field is allowed.

22. (Original) The data structure of claim 19, wherein further, completion of the error handling routine is independent of completion of the staging of other data structures.

23. (Original) A computer program product of a computer readable medium usable with a programmable computer, the computer program product having computer-readable code embodied therein for initializing a storage controller, the computer-readable code comprising instructions for:

commencing an initial microcode load (IML) operation;

initializing a parameter in a field of the tracks of copy state data to a first state;

commencing background staging of copy state data tracks from a disk storage device to a memory device;

changing the state of the parameter of a track to a second state;

receiving a request to access a track of copy state data;

if the requested track of copy state data has been staged and the parameter is in the second state, accessing the requested track of copy state data;

if the requested track of copy state data has not been staged and the parameter is in the first state;

issuing a wait command in response to the request to access the track of copy state data;

staging the requested track of copy state data;

revoking the wait command; and

accessing the requested track of copy state data;

completing the staging of the copy state data tracks; and

completing the IML.

24. (Cancelled)

25. (Cancelled)

26. (Amended) The computer program product of claim ~~25~~23, wherein completion of the IML is independent of completion of the staging of all copy state data tracks.